PROPOSED METHOD OF INVESTIGATION

Trident Project 2004-2005

McMath, Stephen S.

January 23, 2004

Contents

1 Background

This research will analyze an algorithm for integer factorization based on the use of continued fractions and quadratic forms, primarily intending to produce a runtime analysis of the algorithm but also proving several valuable results about continued fractions. This paper provides some background to the problem and a description of the algorithm.

There are several different kinds of factorization; this research will focus on integer factorization. Consider an integer N. Factorization is the process of finding integers p and q greater than 1 such that $N = pq$. Complete factorization would require repeating this process for both p and q until all of the remaining factors are prime (i.e irreducible). However, if an algorithm can be developed to quickly factor N into p and q, the same algorithm can be used over again on p and q. For example, it is easy to see that $105 = 5 \cdot 21$ and then repeat to factor 21. From here, you would see that since $21 = 3 \cdot 7$, $105 = 5 \cdot 3 \cdot 7$. Although in this simple case, the complete factorization is easy to find, this task becomes much harder for large numbers. In number theory, the factors of a number dictate many of the characteristics of the number. For example, Euler's ϕ function, which tells how many numbers less than N are **relatively prime**[1] to N, can be directly calculated from the complete factorization. In the example above, there are $(5-1)(3-1)(7-1) = 48$ integers less than 105 that are relatively prime to 105. Also, determining whether a number is a **quadratic residue** (i.e. the square of another number **modulo** N), can be determined directly using Gauss's **quadratic reciprocity law** if the complete factorization of N is known. In the above example, $19^2 = 361 = 46 + 3 \cdot 105$, so that 46 is a quadratic residue modulo 105 with square root 19. One would represent this as $19^2 \equiv 46 \pmod{105}$. With the complete factorization, we can use the law of reciprocity to analyze 46 modulo 3,5, and 7 to determine whether or not 46 is a quadratic residue without actually having to find its square root first [Ga]. Therefore, the ability to factor large numbers has been a focus of research for a variety of theoretical reasons.

One practical application of factorization is **cryptography**. In one system of encrypting a message, the **RSA** system, the cryptographer chooses a number that is the product of two large numbers p and q that are presumed to be prime: $N = pq$. Then an exponent e is chosen such that e is relatively prime to $(p-1)(q-1)$. Although there are several variations, in the normal public key version, N and e are made public. The user of the RSA

[1] Terms in bold are included in the glossary.

system then privately calculates d, the inverse of e modulo $(p-1)(q-1)$. Anyone is able to encrypt something to him by raising blocks of the message to the power e, modulo N: $c \equiv m^e \pmod{N}$, where m is the original message and c is the encrypted message. Then, the recipient is able to decrypt by evaluating $m \equiv c^d \pmod{N}$ [Riv]. As long as someone intercepting the message is unable to factor N, it is usually impossible to obtain d, so that the message cannot be broken. The security of this system and its variations depends highly on whether or not N can be factored [T]. Although fast factorization would be a threat to this system, the advance in number theory produced by fast factorization would likely provide a number of alternative secure systems.

In addition to the potential for alternative secure systems, there is also the possibility that a fast factorization algorithm might not work for all numbers. Therefore, there could be some numbers for which factorization might be easier than others. If there are classes of numbers that a fast factorization algorithm does not work on, this would allow designers of the algorithm to increase their security by relying more on these numbers. Regardless of whether or not the algorithm works for all numbers or provides alternative systems, for security purposes it is necessary to understand the strengths and weaknesses of the system.

Up to now we have referred to fast factorization in general terms, but there are several different ways to classify the speed of an algorithm. Let N be the number to factor. Let $n = \log_2 N$, the number of bits in N. An algorithm's run time is called "exponential" if it increases exponentially with the number of bits n. "Linear" refers to an algorithm where the time increases proportionally to the number of bits[2]. "Polynomial" refers to an algorithm for which the time required is some polynomial function of n. Thus, linear time is a special case of polynomial time. There are some algorithms that fall in between polynomial and exponential time and are referred to as sub-exponential. Currently, the best general purpose factorization algorithm is the general number field sieve, with a runtime of $\exp(\frac{4}{3^{2/3}} n^{1/3} (\log n)^{2/3})$ [L]. The $\frac{1}{3}$ in the exponent of n has a very significant effect on the

[2] Since $n = \log_2 N$, so that such a runtime is logarithmic in N, this is often referred to as logarithmic, resulting in a certain amount of confusion.

runtime of the algorithm, as this determines that the algorithm is sub-exponential.

Many of the other factorization algorithms are important for theoretical reasons. One tool used by several different algorithms is the continued fraction expression for $\sqrt{N}$, where N is the number to be factored. This expression is calculated recursively:

$$x_0 = \sqrt{N},\ b_0 = \lfloor x_0 \rfloor \ \ (\textit{the } \textbf{floor} \textit{ of } x_0).$$

$$\forall i \geq 1\ x_i = \frac{1}{x_{i-1} - b_{i-1}} \quad b_i = \lfloor x_i \rfloor \tag{1}$$

$$\sqrt{N} = b_0 + \frac{1}{b_1 + \frac{1}{b_2 + \ldots}}$$

It is important to note that at each step in the continued fraction expansion, it is possible to define integers P_i and Q_i such that $x_i - b_i = \frac{\sqrt{N} - P_i}{Q_i}$, where $P_i^2 \equiv N \pmod{Q_i}$. Also, in the expansion of $\sqrt{N}$, $\forall i$, $(-1)^i Q_i Q_0$ is a quadratic residue modulo N. Both of these were proven in Hans Riesel [Rie] and are included in Theorem 1 of section 6. The sequence of Q_i's are thus referred to as **pseudo-squares**. The sequence of P_i's are referred to as **residues**, since they are produced at each step by reducing the fraction by some integer. Several algorithms, most notably the Morrison - Brillhart algorithm, rely on the quadratic residues in the sequence of pseudo-squares. In 1982 Daniel Shanks developed but never published[3] an algorithm called Square Forms Factorization, or SQUFOF, which takes a more direct approach to using these quadratic residues.

[3] He did however refer to it several times. See p 172 of [S].

```
$x_0 \leftarrow \sqrt{N}$ $b_0 \leftarrow \lfloor x_0 \rfloor$
while $Q_i \neq$ perfect square
    Apply equation(1)
    Reduce $x_i$ to the form $\frac{\sqrt{N}-P_i}{Q_i}$
$x'_0 \leftarrow \frac{\sqrt{N}+P_i}{\sqrt{Q_i}}$
while $P_i \neq P_{i-1}$
    Apply equation (1) and reduce

gcd($N, P_i$), the greatest
common divisor, is a nontrivial
factor.
```

Figure 1: SQUFOF algorithm

For SQUFOF, when the algorithm encountered a **perfect square**, it started a new sequence from that term by taking the **conjugate** of the numerator and the square root of the denominator and then continued the expansion until a residue repeated consecutively (Figure 1). At this point, the repeated residue provided a factor of N.

For example, let N be 1353:

$$x_0 = \sqrt{1353} \; b_0 = 36$$

$$x_1 = \frac{1}{\sqrt{1353}-36} = \frac{\sqrt{1353}+36}{57} = 1 + \frac{\sqrt{1353}-21}{57}$$

$$x_2 = \frac{57}{\sqrt{1353}-21} = \frac{\sqrt{1353}+21}{16} = 3 + \frac{\sqrt{1353}-27}{16} \qquad (2)$$

The second fraction in each step is found by rationalizing. At each step, the integers taken out are b_i and the remaining fractions are between 0 and 1. After subtracting b_i the remaining fraction is inverted to find x_{i+1}. As a point of reference, we have approximated so far that $\sqrt{1353} \approx 36 + \frac{1}{1+\frac{1}{3}}$. SQUFOF stops here because 16 is a perfect square. Taking the conjugate of the top and the square root of the bottom, we obtain:

$$x'_0 = \frac{\sqrt{1353}+27}{4} = 15 + \frac{\sqrt{1353}-33}{4}$$

$$x'_1 = \frac{4}{\sqrt{1353}-33} = \frac{\sqrt{1353}+33}{66} = 1 + \frac{\sqrt{1353}-33}{66} \qquad (3)$$

Here, since the residue 33 is repeated, we quickly find that 33 is a factor of 1353. $1353 = 33 \cdot 41 = 3 \cdot 11 \cdot 41$. The explanation of this can be seen by applying cross multiplication of the two fractions on the left of (3) and simplifying. This produces the equation $N - P_i^2 =$

$1353 - 33^2 = 4 \cdot 66 = Q_i Q_{i+1}$, as in (a) of Theorem 1 in section 6. Since the residue $P_i = P_{i+1} = 33$ is repeated, it must have a factor in common with the pseudo-square $Q_{i+1} = 66$, because of the recurrence relation $P_{i+1} = b_{i+1}Q_{i+1} - P_i$ in (b) of Theorem 1 below. This common factor must then also divide 1353.

Shanks also showed that the second part of SQUFOF (the process of finding a factorization once the perfect square has been found) is linear using a process called composition of quadratic forms. Composition of quadratic forms has several different definitions. Originally, Gauss defined it as the process of multiplying two quadratic forms together and making a substitution to reduce the product to another quadratic form [Ga]. Therefore, I will use the multiplication symbol $(*)$ for composition. However, this computation is slow and complicated. Shanks developed an approximation to this that relies on an extended **Euclidean algorithm** and the Chinese remainder theorem (Figure 2). This is much faster and simpler and is useful for a variety of theoretical reasons. Shanks also developed two algorithms for performing a partial reduction before the composition was completed: NUCOMP and NUDUPL. NUCOMP composes two different quadratic forms while NUDUPL composes a quadratic form with itself [S]. These algorithms, although slightly faster, are too complicated to explain here.

```
$(a, b, -c)$ is the quadratic form
        to be composed with itself
$(a', b', -c')$ is the result
$a' \leftarrow a^2$
$f \leftarrow \frac{b^2-N}{2 \cdot b}$
Reduce $f$ to least terms $\frac{p}{q}$
$b' \leftarrow b - f \pmod{a'}$
$c' = \frac{N-b^2}{a}$

Result: $(a', b', -c')$
```

Figure 2: Simple variation of composition of quadratic forms

From the second line of (2), observe that $21^2 + 57 \cdot 16 = 1353$. The quadratic form for this step would be $57x^2 + 2 \cdot 21xy - 16y^2$, abbreviated $(57, 21, -16)$. In general, the quadratic form for each step is $(Q_{i-1}, P_i, -Q_i)$. Given a quadratic form $(a, b, -c)$, setting $x_0' = \frac{\sqrt{N}-b}{a}$ returns to the continued fraction expansion. The result of composition is another quadratic form similarly related to somewhere else in the continued fraction expansion. Regardless of the composition algorithm chosen, the results are within several steps of each other in the continued fraction expansion.

The results of composition are still often outside the bounds normally obtained in the continued fraction expansion, so it is often necessary to reduce this product further. Shanks followed an algorithm designed by Gauss that uses substitutions. However, this algorithm also sometimes reverses the direction of the quadratic form, a trait that is undesirable for this research for reasons that will become apparent later, so quadratic forms will instead be converted back into a step in the continued fraction expansion in order to reduce them. For example, if we compose $(57, 21, -16)$ with itself using NUDUPL, we obtain $(31, 50, 37)$. In order to reduce this, we write out the step in the expansion that it would represent and proceed from there:

$$\frac{31}{\sqrt{1353}-50} = \frac{\sqrt{1353}+50}{-37} = -3 + \frac{\sqrt{1353}-61}{-37}$$

$$\frac{-37}{\sqrt{1353}-61} = \frac{\sqrt{1353}+61}{64} = 1 + \frac{\sqrt{1353}-3}{64}$$

$$\frac{64}{\sqrt{1353}-3} = \frac{\sqrt{1353}+3}{21} = 1 + \frac{\sqrt{1353}-18}{21}$$

$$\frac{21}{\sqrt{1353}-18} = \frac{\sqrt{1353}+18}{49} = 1 + \frac{\sqrt{1353}-31}{49}$$

In the fourth step, $0 < 49 < 2\sqrt{N} \approx 73$, $0 < 18 < \sqrt{N} \approx 36$, and $\sqrt{N} - 18 < 49$, so that the quadratic form is completely reduced[4]. The new quadratic form is found from the fourth step to be $(21, 18, -49)$. This same step could be found in the 7th step of the continued fraction expansion if composition of quadratic forms were not used. The number of terms skipped is larger for a quadratic form farther down in the expansion and can be roughly approximated, so that if it is known which step in the process is desired, composition of quadratic forms gets close enough that the answer can be quickly found. Although the first

[4] As described in Theorem 1, the conditions are: $0 < Q_i < 2\sqrt{N}$, $0 < P_{i-1} < \sqrt{N}$, and $\sqrt{N} - P_{i-1} < Q_i$.

phase of Shanks' SQUFOF algorithm, the process of finding a perfect square, still requires exponential time, the total time is roughly cut in half [Rie].

2 Current Research

There are a few important observations about Shanks' SQUFOF algorithm. First, in the original continued fraction expansion, we started with pseudo-square $Q_0 = 1$. However, it is possible to start with any integer Q_0 such that N is a quadratic residue of Q_0, so that P_0 can be found such that $P_0^2 \equiv N \pmod{Q_0}$. If we choose $Q_0 = 2$, and $P_0 = \left\lfloor \sqrt{N} \right\rfloor$ or $\left\lfloor \sqrt{N} \right\rfloor - 1$, such that P_0 is odd, then we have $x_0 - b_0 = \frac{\sqrt{N} - P_0}{Q_0}$. In Theorem 2 of section 6, I prove that if $N \equiv 1 \pmod 4$ and -1 is not a quadratic residue, then this sequence of pseudo-squares and residues will provide a factorization of N. However, this is not a complete description of the numbers that this sequence provides a factorization for. For example, for $N = 35$, a factorization is immediate when we evaluate $\lfloor \sqrt{N} \rfloor = 5$, even though $35 \equiv 3 \pmod 4$ and 377, for which a factor of 13 is found on the 2nd step but for which -1 is a quadratic residue.

$$N = 1333 \quad \left\lfloor \sqrt{1333} \right\rfloor = 36$$
$$Q_0 \leftarrow 2 \quad P_0 \leftarrow 35$$
$$\frac{2}{\sqrt{1333}-35} = \frac{\sqrt{1333}+35}{54} = 1 + \frac{\sqrt{1333}-19}{54}$$
$$\frac{54}{\sqrt{1333}-19} = \frac{\sqrt{1333}+19}{18} = 3 + \frac{\sqrt{1333}-35}{18}$$
$$\frac{18}{\sqrt{1333}-35} = \frac{\sqrt{1333}+35}{6} = 11 + \frac{\sqrt{1333}-31}{6}$$
$$\frac{6}{\sqrt{1333}-31} = \frac{\sqrt{1333}+31}{62} = 1 + \frac{\sqrt{1333}-31}{62}$$

Since the residue 31 is repeated,
it is a factor. $1333 = 31 \cdot 43$.

Figure 3: example of first shortcut

Figure 3 provides an example of applying this shortcut normally. Although performing SQUFOF without the shortcut would take about the same number of steps in this example, the time saved is more significant for many other numbers. Also observe that if we continue the expansion after the factorization is obtained, the sequences repeat, except in the opposite order and paired differently (Figure 4). After inverting, this last fraction will be the same as the fraction we started with, so the entire process repeats from here. Lemma 3 defines this symmetry more explicitly.

Based on this symmetry, any fast test to determine whether or not the continued fraction expansion is in the correct direction, that is, whether or not it has not passed the factorization yet, would provide a faster factorization algorithm by performing a binary search (Figure 5).

$$\begin{aligned}
\frac{62}{\sqrt{1333}-31} &= \frac{\sqrt{1333}+31}{6} = 11 + \frac{\sqrt{1333}-35}{6}\\
\frac{6}{\sqrt{1333}-35} &= \frac{\sqrt{1333}+35}{18} = 3 + \frac{\sqrt{1333}-19}{18}\\
\frac{18}{\sqrt{1333}-19} &= \frac{\sqrt{1333}+19}{54} = 1 + \frac{\sqrt{1333}-35}{54}\\
\frac{54}{\sqrt{1333}-35} &= \frac{\sqrt{1333}+35}{2} = 35 + \frac{\sqrt{1333}-31}{2}
\end{aligned}$$

Figure 4: example of symmetry

```
Given $N$ to be factored:
Verify conditions on $N$.
$Q_0 \leftarrow 2$, $P_0 \leftarrow \lfloor\sqrt{N}\rfloor$ or $\left\lfloor\sqrt{N}\right\rfloor - 1$, such that $P_0$ is odd.
Apply equation (1) for 2 steps.
$F_0 \leftarrow$ quadratic form from 2nd step of continued fraction expansion
$i \leftarrow 0$
while $F_i$ is in the right direction
    $F_{i+1} \leftarrow F_i * F_i$
    $i \leftarrow i + 1$
$F_{last} \leftarrow F_{i-1}$
while $i \geq 0$
    if $F_{last} * F_i$ is in the right direction
        $F_{last} \leftarrow F_{last} * F_i$
    $i \leftarrow i - 1$
Convert $F_{last}$ to a continued fraction step and use equation 1
        to expand several steps to obtain the factorization.
```

Figure 5: intended algorithm using quadratic forms to perform a binary search of the continued fraction expansion

I provide one example of this algorithm:

$N = 2035153$, $Q_0 = 2$, $P_0 = 1425$

Letting the indices refer to the related step in the standard continued fraction expansion we obtain[5]: $F_2 = (1132, 839, -294)$.

We proceed by using NUDUPL to perform composition of quadratic forms, reducing the results by use of the continued fraction expansion.

[5] Due to the change from $Q_0 = 1$ to $Q_0 = 2$, the general form for the quadratic form is now $(Q_{i-1}/2, P_i, -Q_i/2)$

$$F_2 * F_2 = (696, 1399, -28) = F_{11}$$

$$F_{11} * F_{11} = (81, 1403, -206) = F_{27}$$

$$F_{27} * F_{27} = (739, 545, -588) = F_{55}$$

$$F_{55} * F_{55} = (696, 457, -656) = F_{119}$$

$$F_{119} * F_{119} = (576, 1207, -251), \text{ which is in the wrong direction.}$$

$$F_{119} * F_{55} \text{ and } F_{119} * F_{27} \text{ are reversed.}$$

$$F_{119} * F_{11} = (24, 1415, -343) = F_{135}$$

$$F_{135} * F_2 = (246, 1019, -1013) = F_{140}$$

Converting back to continued fractions:

$$\frac{2 \cdot 246}{\sqrt{N} - 1019} = \frac{\sqrt{N} + 1019}{2026} = 1 + \frac{\sqrt{N} - 1007}{2026}$$

$$\frac{2026}{\sqrt{N} - 1007} = \frac{\sqrt{N} + 1007}{504} = 4 + \frac{\sqrt{N} - 1009}{504}$$

$$\frac{504}{\sqrt{N} - 1009} = \frac{\sqrt{N} + 1009}{2018} = 1 + \frac{\sqrt{N} - 1009}{2018}$$

we obtain the factor 1009: $2035153 = 1009 \cdot 2017$

The decisions for this example of whether or not a form was reversed were determined by merely comparing with the actual continued fraction expansion. However, ideally this can be done without doing the entire expansion.

I conjecture based on empirical evidence that in the process of expanding the continued fraction expansion that if $Q_i | Q_{i-1}$, $(Q_i)^3 \nmid Q_{i-1}$, and Q_i is not a power of 2, then the continued fraction expansion is in the correct direction, that is, it has not passed the factorization yet. If $(Q_i)^3 | Q_{i-1}$ or Q_i is a power of 2, no information is provided. As an intuitive explanation of where this conjecture comes from, observe that this always happens in Shanks' SQUFOF algorithm if you reverse the step (same as not taking the conjugate of the numerator according to Lemma 2). In the first example, if we had ignored 16 we would have found 49 on the 8th step, with $x_8 - b_8 = \frac{\sqrt{1353}-31}{49}$. Taking the square root of 49 and not taking the conjugate of the numerator, and then inverting, we obtain $\frac{7}{\sqrt{1353}-31} = \frac{\sqrt{1353}+31}{56}$, so that $Q'_0 = 7$, $Q'_{-1} = 56$ and $7 \mid 56$.

The condition that $Q_i | Q_{i-1}$ is extremely rare, but it is possible to find a multiple k such that in the expansion with Q_i unchanged, P_i replaced by kP_i and N replaced by k^2N, this condition is met. For notation, let $Q_i = Q'_0$. Since using $-k$ instead of k produces the same sequence in the opposite direction, it is necessary to relate this sequence to the original sequence in order to extract useful information. From this, I have found so far that if $Q_{i+3}|Q'_1$ and $Q'_0|Q'_{-1}$, then the original expansion is in the correct direction. Finding multiples such that the second condition is satisfied can be accomplished by using continued fractions to find the convergents of $\frac{\sqrt{N}-R_i}{Q_i^2}$, where $R_i \equiv \sqrt{N} \equiv P_i - \frac{P_i^2-N}{2P_i} \pmod{Q_i^2}$.

Although most of the multiples do not cause the sequence to satisfy the first condition, it empirically appears to be possible roughly 2% of the time, regardless of the size of N. If this is true, even this crude result could provide a polynomial time factorization algorithm, a very significant discovery, by merely attempting the test an average of 50 times for each test. Thus, the runtime would be roughly $c(\log N)^{c'}$. However, polynomial time factorization could be possible even without this frequency of response, since all that is required for polynomial time factorization is that the time required for this test of direction be a polynomial function of the number of digits. In other words, the frequency with which the test provides an answer needs to be bounded below by the reciprocal of a polynomial. Even if this were not possible, it is hopeful that the algorithm would provide a valuable tool to number theory.

3 Proposed Research

My primary goal is to obtain an analysis of the average case runtime. However, in order to obtain this, I will need some preliminary results.

First, I intend to investigate more fully the conditions for which the sequence of pseudo-squares and residues will provide a factorization. A set of numbers that I can prove this algorithm does not work on would be extremely valuable for security purposes. Second, I intend to investigate the connection with quadratic forms more fully. Although Shanks assumed it to be true and probably had a proof, I have not yet seen a formal proof that the composition of two quadratic forms is another quadratic form in the same continued fraction expansion, an especially interesting fact considering the usual necessity of reducing the result. Third, I intend to investigate the conjecture that the test of direction is accurate. If it were possible to prove these, the proofs would provide valuable information for assessing the algorithm as it currently stands, provide valuable insight into possible improvements, and provide a base of information about continued fractions that someone else could build on later.

Second, I intend to analyze the ability to perform a test of direction, specifically whether the frequency changes with larger choices of N and how the time required for this test changes. This combined with proofs or at least good arguments for the other conjectures will provide for an analysis of average runtime. This will determine how serious of a threat it poses to the RSA system. I intend to use C++ both to research the algorithm and to produce a working implementation of the algorithm.

4 Proposed Timeline

February 2004 - April 2004

I will investigate the conditions for which this algorithm provides a factorization. In Theorem 2, I have already proved that the algorithm works for a significant number of integers[6], but I still need to research more fully what integers it does and does not work on and a more thorough explanation of why. I will continue researching the work done by Daniel Shanks, specifically attempting to obtain a copy of an incomplete paper in which he describes SQUFOF in more detail [W], hopefully explaining in some detail where he originally got his

[6] Integers N such that $N \equiv 1 \pmod 4$, but such that -1 is not a quadratic residue compose at least 1/4 of the non-prime odd integers.

idea and why it works. This should provide some light into the rest of the project.

May 2004 - July 2004

I will investigate the connection between quadratic forms and continued fractions. I will begin by seeking for some characteristic of a quadratic form that distinguishes the continued fraction that it occurs in. In addition to providing a formal proof that a binary search is possible, this research is likely to also shed some light on the test of direction itself.

August 2004 - October 2004

I will investigate the conjecture that the first part of the test of direction is accurate, the relatively simple case when the condition $Q_i|Q_{i-1}$ is met. I will first search for counterexamples. If I am able to find any, I will analyze what goes wrong. From this, I will modify the conditions if necessary. I will also investigate the connection back to SQUFOF. Either way, I believe that a proof by induction on i, the index of the pseudo-squares, should be possible. In mid October, I will begin work on the mid-term report in order to finish it by the beginning of December.

November 2004 - December 2004

I will investigate the second part of the test of direction, the case when a multiple k is used and the condition that $Q_{i+3}|Q'_1$ is met. I will first analyze to what extent Q_{i+5} and Q_{i+7} relate similarly. This will either provide an insight into how Q_{i+3} is distinct or how the entire sequence may be related to the original sequence. I will attempt a proof, but I believe that in the process of developing a proof, I will discover that it is possible to achieve this test some other way.

In December, I will finish the mid-term report and begin working on the final write-up.

January 2005 - February 2005

I will analyze the frequency with which a test of direction may be performed. At this point, it is possible that the approach to testing direction may be entirely changed by discoveries through the earlier proofs. However, regardless of how the test is done, I will first use significant computer time conducting this analysis. From this, I will analyze which numbers have a higher or lower frequency and attempt to understand the variations. Also, I will attempt to understand the characteristics of pseudo-squares that do provide for a test of direction. With this understanding, I should be able to produce proof of this frequency, which will then provide a complete analysis of runtime.

I will continue inserting this information into my final write-up.

March 2005 - May 2005

I will complete the final write-up for the Trident Scholar Committee and produce a running computer implementation of the algorithm.

5 Glossary

Conjugate: changes the sign of a certain term in an expression, traditionally the part of an expression that is either irrational or imaginary. For this research, it applies to changing the sign of a residue, the integer part of the numerator of an x_i in the continued fraction expansion.

Cryptography: set of methods for encrypting information to prevent it from being read by anyone who intercepts the messege. Used in variety of civil and military applications.

Euclid's Algorithm: fast algorithm for determining the greatest common divisor of two integers. Given x and y, the extended Euclidian algorithm also determines the coefficients a and b such that $ax + by = \gcd(x, y)$.

Floor: greatest integer less than or equal to a given number. Symbol: $\lfloor x \rfloor$

Greatest Common Divisor: largest integer that divides a pair or group of integers. Symbol: $gcd(x, y)$

Modular Arithmetic: two numbers are considered equal (congruent) if heir difference is divisible by the base. Thus, $3 \equiv 10 \pmod 7$. Numbers are represented by integers between 0 and $N - 1$, where N is the base. Multiplication, addition, and subtraction are normal, except that the results are reduced. Division is performed by reducing fractions to least terms, applying an extended Euclid's algorithm to find the inverse of the denominator, and then performing multiplication. Symbol: (mod N)

Modulo: operation related to division that returns the remainder:

$\frac{73}{11} = 6\frac{7}{11}$, so 73 modulo 11 = 7. Symbol: % or (mod N)

NUCOMP-NUDUPL: algorithms designed by Daniel Shanks to perform composition of quadratic forms quickly.

Perfect Square: integer that is the square of another integer. Thus, 9 is a perfect square because $3^2 = 9$.

Pseudo-square: the denominator of an x_i in the continued fraction expansion, denoted Q_i. When $Q_0 = 1$, $(-1)^i Q_i$ is a quadratic residue and in general $-Q_i Q_{i-1}$ is a quadratic residue.

Quadratic Reciprocity Law (Gauss): determines which numbers are quadratic residues of a prime:

Symbol: $(\frac{a}{p}) = 1$ if $x^2 \equiv a \pmod{p}$ has a solution, -1 if it does not, and 0 if $p \mid a$.

Theorem: For p and q distinct primes:

$$(\frac{p}{q})(\frac{q}{p}) = (-1)^{(p-1)(q-1)/4}$$

$$(\frac{2}{p}) = (-1)^{(p^2-1)/8}$$

$$(\frac{-1}{p}) = (-1)^{(p-1)/2}$$

Quadratic Residue: a perfect square modulo some base N. 2 is a quadratic residue of 7 because $3^2 \equiv 2 \pmod{7}$.

Relatively Prime: having no common divisors. Thus, 8 and 15 are relatively prime, even though they are not prime by the normal definition.

Residue: integer that remains in the numerator after b_i has been subtracted from x_i in the continued fraction expansion.

RSA: cryptology algorithm named after Rivest, Shamir, and Adleman. It was earlier developed by Clifford Cooks of GCHQ, but this was only recently declassified. Its security is dependent on the difficulty of factorization [E].

SQUFOF: Square Forms Factorization, developed by Daniel Shanks in 1982.

6 Proofs

Throughout, we will assume that N, the odd positive integer to be factored, is not a perfect square and that $N \equiv 1 \pmod 4$. Also, we are assuming that $x_0 = \frac{\sqrt{N}+P_0}{2}$, where $P_0 = \lfloor\sqrt{N}\rfloor$ or $\lfloor\sqrt{N}\rfloor - 1$, such that P_0 is odd. Note that with this definition, $0 < x_0 - P_0 < 1$, so that $P_0 = b_0$. Also note that we have defined by this that $Q_0 = 2$. The recursive formulas are:

$$x_{i+1} = \frac{1}{x_i - b_i} \quad b_i = \lfloor x_i \rfloor,\ i \geq 0$$

Formally, the equation we are assuming is:

$$x_{i+1} = \frac{Q_i}{\sqrt{N} - P_i} = \frac{\sqrt{N} + P_i}{Q_{i+1}} = b_{i+1} + \frac{\sqrt{N} - P_{i+1}}{Q_{i+1}},\ i \geq 0 \tag{4}$$

Note that this equation serves as a definition of Q_i, P_i, Q_{i+1}, and P_{i+1}, so that these equations are true regardless of the conditions on these variables.

Theorem 1 *In the continued fraction expansion, each x_i reduces to the form $\frac{\sqrt{N}+P_{i-1}}{Q_i}$, with (a) $N = P_i^2 + Q_iQ_{i+1}$, (b) $P_i = b_iQ_i - P_{i-1}$, (c) $b_i > 0$, (d) $0 < P_i < \sqrt{N}$, (e) $0 < Q_i < 2\sqrt{N}$, (f) Q_i is an integer. Furthermore, (g) this sequence is eventually periodic [Rie].*

Proof:

(a) From (4), the equation $\frac{Q_i}{\sqrt{N}-P_i} = \frac{\sqrt{N}+P_i}{Q_{i+1}}$ requires that $N = P_i^2 + Q_iQ_{i+1}$.

(b) It is evident from simplifying the expression on the far right of 4 that $\frac{\sqrt{N}+P_i}{Q_{i+1}} = \frac{\sqrt{N}+b_{i+1}Q_{i+1}-P_{i+1}}{Q_{i+1}}$. Therefore, we have that $P_i = b_iQ_i - P_{i-1}$.

(c) For $i = 0$, $b_0 = P_0 > 0$.

For $i > 0$, $b_{i-1} = \lfloor x_{i-1} \rfloor$. By the definition of floor, we then have that $x_{i-1} - 1 < b_{i-1} \leq x_{i-1}$. If $b_{i-1} = x_{i-1}$, then the continued fraction expression formed by the sequence (b_i) is a rational expression equal to $x_0 = \frac{\sqrt{N}+P_0}{2}$, which is irrational since N is not a perfect square, providing a contradiction. Therefore, $x_{i-1} - 1 < b_{i-1} < x_{i-1}$, so that $0 < x_{i-1} - b_{i-1} < 1$. Therefore, $x_i = \frac{1}{x_{i-1}-b_{i-1}} > 1$, so that $b_i = \lfloor x_i \rfloor \geq 1 > 0$.

(d-e) I will prove inductively that $0 < Q_i < 2\sqrt{N}$ and $0 < P_{i-1} < \sqrt{N}$.

Base case: $i = 1$

$P_0 = \lfloor \sqrt{N} \rfloor$ or $\lfloor \sqrt{N} \rfloor - 1$, so by definition $0 < P_0 < \sqrt{N}$.

$Q_1 = \frac{N-P_0^2}{2} > 0$. Also, $1 = \frac{N-P_0^2}{2Q_1} = \frac{\sqrt{N}-P_0}{2}\frac{\sqrt{N}+P_0}{Q_1}$. Since $\frac{\sqrt{N}-P_0}{2} < 1$, we must have $\frac{\sqrt{N}+P_0}{Q_1} > 1$, so that $Q_1 < \sqrt{N} + P_0 < 2\sqrt{N}$.

Induction: Assume $0 < Q_i < 2\sqrt{N}$ and $0 < P_{i-1} < \sqrt{N}$.

From (c), $0 < x_i - b_i < 1$ means $0 < \frac{\sqrt{N}-P_i}{Q_i} < 1$. Since $Q_i > 0$, we can say $0 < \sqrt{N} - P_i < Q_i$. From the left side of this, we have that $P_i < \sqrt{N}$. Now, either $Q_i \leq \sqrt{N}$ or $Q_i > \sqrt{N}$.

Case 1: If $Q_i \leq \sqrt{N}$, then $\sqrt{N} - P_i < Q_i \leq \sqrt{N}$, so that $P_i > 0$.

Case 2: If $Q_i > \sqrt{N}$, then by (b), $P_i = b_i Q_i - P_{i-1} > b_i\sqrt{N} - \sqrt{N} = (b_i - 1)\sqrt{N} > 0$.

Therefore, $0 < P_i < \sqrt{N}$.

By (a), $Q_{i+1} = \frac{N-P_i^2}{Q_i}$. Since $0 < P_i < \sqrt{N}$, $N - P_i^2 > 0$, so since also $Q_i > 0$, we have that $Q_{i+1} > 0$.

$1 = \frac{N-P_i^2}{Q_iQ_{i+1}} = \frac{\sqrt{N}-P_i}{Q_i}\frac{\sqrt{N}+P_i}{Q_{i+1}}$. Since $\frac{\sqrt{N}-P_i}{Q_i} < 1$, we must have that $\frac{\sqrt{N}+P_i}{Q_{i+1}} > 1$. Since $Q_{i+1} > 0$, this implies $Q_{i+1} < \sqrt{N} + P_i < 2\sqrt{N}$.

(f) The fact that $N = P_i^2 + Q_iQ_{i+1}$ requires that $Q_{i+1} = \frac{N-P_i^2}{Q_i}$. In order to show that $\forall i$ Q_i is an integer, I will prove by induction that Q_i is an integer and $Q_i \mid N - P_i^2$.

Base case: $i = 0$

N and P_0 are odd by their definitions, so $N - P_0^2$ is even, so that $2 \mid N - P_0^2$. But $Q_0 = 2$, so the statement is true for $i = 0$.

Induction: Assume for some i, Q_i is an integer and $Q_i \mid N - P_i^2$. Then, since $N =$

$P_i^2 + Q_iQ_{i+1}$, $Q_{i+1} = \frac{N-P_i^2}{Q_i}$, so that since $Q_i \mid N - P_i^2$, Q_{i+1} is an integer. Also, $Q_i = \frac{N-P_i^2}{Q_{i+1}}$, so that since Q_i is an integer, $Q_{i+1} \mid N - P_i^2$. Since $P_{i+1} = b_{i+1}Q_{i+1} - P_i$,

$$N - P_{i+1}^2 = N - (b_{i+1}Q_{i+1} - P_i)^2 = N - b_{i+1}^2Q_{i+1}^2 + 2b_{i+1}Q_{i+1}P_i - P_i^2$$

$$= (N - P_i^2) - b_{i+1}^2Q_{i+1}^2 + 2b_{i+1}Q_{i+1}P_i$$

Since $Q_{i+1} \mid N - P_i^2$ and $Q_{i+1} \mid -b_{i+1}^2Q_{i+1}^2 + 2b_{i+1}Q_{i+1}P_i$, we have that $Q_{i+1} \mid N - P_{i+1}^2$ and the induction is complete.

(g) Since each x_i and thus the entire sequence that follows it is defined by the two integers Q_i and P_{i-1}, limited by the bounds $0 < Q_i < 2\sqrt{N}$ and $0 < P_i < \sqrt{N}$, there is only a finite number of distinct x_i's. Therefore, for some m and some k, $\forall i \geq k$ $x_i = x_{i+m}$. QED

Lemma 1

$$\lfloor \frac{\sqrt{N} + P_i}{Q_i} \rfloor = \lfloor \frac{\sqrt{N} + P_{i-1}}{Q_i} \rfloor = b_i$$

Proof: The second part of this equation, that $\lfloor \frac{\sqrt{N}+P_{i-1}}{Q_i} \rfloor = b_i$ follows from the definition of b_i.

In order to show that $\lfloor \frac{\sqrt{N}+P_i}{Q_i} \rfloor = \lfloor \frac{\sqrt{N}+P_{i-1}}{Q_i} \rfloor$, I will first show that

$$\frac{\sqrt{N} + P_i}{Q_i} > 1.$$

Assume the contrary, that $Q_i \geq \sqrt{N} + P_i$. Then,

$$b_i(\sqrt{N} + P_i) - P_i \leq b_iQ_i - P_i = P_{i-1} < \sqrt{N},$$

$$b_i\sqrt{N} + P_i(b_i - 1) < \sqrt{N},$$

$$\sqrt{N}(b_i - 1) + P_i(b_i - 1) < 0,$$

$$(b_i - 1)(\sqrt{N} + P_i) < 0.$$

But $\sqrt{N}$ and P_i are positive, so this implies $b_i < 1$, contradicting the fact that $b_i > 0$, since b_i must be an integer. Therefore,

$$Q_i < \sqrt{N} + P_i, \quad \frac{\sqrt{N} + P_i}{Q_i} > 1$$

From this, we find that $Q_{i+1} = \frac{\sqrt{N} - P_i^2}{Q_i} = \frac{(\sqrt{N}+P_i)(\sqrt{N}-P_i)}{Q_i} > \sqrt{N} - P_i$. Therefore,

$$\lfloor \frac{\sqrt{N} + P_i}{Q_i} \rfloor = \lfloor \frac{\sqrt{N} + b_i Q_i - P_{i-1}}{Q_i} \rfloor = b_i + \lfloor \frac{\sqrt{N} - P_{i-1}}{Q_i} \rfloor = b_i. \ QED$$

Lemma 2 *If $x_i - b_i = \frac{\sqrt{N}-P_i}{Q_i}$ and $x_{i+1} = \frac{1}{x_i - b_i} = b_{i+1} + \frac{\sqrt{N}-P_{i+1}}{Q_{i+1}}$ and if we assign $y_0 = \frac{\sqrt{N}+P_{i+1}}{Q_{i+1}}$, then $c_0 = \lfloor y_0 \rfloor = b_{i+1}$ and $y_1 = \frac{1}{y_0 - c_0} = \frac{\sqrt{N}+P_i}{Q_i}$*

Proof: By Lemma 1, $c_0 = \lfloor y_0 \rfloor = \lfloor \frac{\sqrt{N}+P_{i+1}}{Q_{i+1}} \rfloor = b_{i+1}$

$$y_1 = \frac{1}{y_0 - c_0} = \frac{1}{\frac{\sqrt{N}+P_{i+1}}{Q_{i+1}} - b_{i+1}} = \frac{1}{\frac{\sqrt{N}+P_{i+1}-b_{i+1}Q_{i+1}}{Q_{i+1}}}$$

$$= \frac{1}{\frac{\sqrt{N}-P_i}{Q_{i+1}}} = \frac{\sqrt{N} + P_i}{\frac{N-P_i^2}{Q_{i+1}}} = \frac{\sqrt{N} + P_i}{Q_i} \ QED$$

This demonstrates an important fact about continued fractions, the fact that the direction of the sequences of pseudo-squares and residues can be reversed (i.e the indices decrease) by taking the conjugate and applying the same recursive mechanism. Thus, if the starting condition near some point is the same in both directions, the sequence will be symmetric about that point. This is the point of Lemma 3. Note that this lemma implicitly uses the fact there is only one odd integer P_0 such that $0 < \sqrt{N} - P_0 < 2$

Lemma 3 *Let negative indices represent pseudo-squares found using the reversal of direction* *defined in Lemma 2. The sequence of pseudo-squares is symmetric about $Q_0 = 2$, so that, using this notation $\forall i\ Q_i = Q_{-i}$.*

Proof: Let $y_{-1} = \frac{\sqrt{N}+P_1}{Q_1}$. Then, by Lemma 2, $y_0 = \frac{\sqrt{N}+P_0}{Q_0} = \frac{\sqrt{N}+P_0}{2}$

However, this is the same as x_0, so the sequence of pseudo-squares will be symmetric about $Q_0 = 2$. Therefore, $Q_i = Q_{-i}$. QED

Combining periodicity with reversibility, we can make a slightly stronger statement about periodicity.

Lemma 4 *There exists a positive integer m such that $\forall i\ x_i = x_{i+m}$, i not necessarily positive.*

Proof: Essentially, I need to prove that in Theorem 1 (g), there is no lower bound for k. Assume the contrary, that there is some lower bound k. Let m and k as in Theorem 1 (g) such that m is the smallest such positive integer and k is the smallest such integer, assuming it exists. Then $x_k = x_{k+m}$. But by Lemma 2 we have that $x_{k-1} = x_{k+m-1}$, so that $k-1$ also meets this criteria, violating the assumption that k is the smallest such integer. Therefore, $\forall i\ x_i = x_{i+m}$. QED

Based on the symmetry about $Q_0 = 2$, we are able to show that there is another point of symmetry.

Lemma 5 *Let $s = \lfloor \frac{m}{2} \rfloor$, where m is the period from Lemma 4. If m is even, $\forall i\ Q_{s+i} = Q_{s-i}$, but $Q_s \neq 2$. If m is odd, $\forall i\ Q_{s+i+1} = Q_{s-i}$.*

Proof:

Case 1: If m is even, $m = 2s$. Then, by Lemmas 3 and 4, $Q_{s+i} = Q_{-s-i} = Q_{2s-s-i} = Q_{s-i}$.

If $Q_s = 2$, $x_s = \frac{\sqrt{N}+P_0}{2}$ since P_0 is the only odd integer such that $0 < \sqrt{N} - P_0 < 2$, so that s is a shorter period than m, contradicting the fact that m is the smallest positive integer such that $\forall i\ Q_i = Q_{i+m}$. Therefore, $Q_s \neq 2$.

Case 2: If m is odd, $m = 2s + 1$. Then, by Lemma 3 and 4, $Q_{s+i+1} = Q_{-s-i-1} = Q_{2s+1-s-i-1} = Q_{s-i}$. QED

The symmetry about this other point provides the mechanism for being able to find this point. Theorem 2 provides the importance of being able to find this point.

Lemma 6 *$\forall i$ Q_i is even.*

Proof: I will prove by induction that if α is chosen such that $2^\alpha \parallel Q_i$, then $2^{\alpha+1} \mid N - P_i^2$.

Base case: $i = 0$

$2^1 \parallel Q_0$, so $\alpha = 1$.

$N \equiv 1 \pmod 4$ and P_0 is odd, so $4 \mid N - P_0^2$, but then $4 = 2^2 = 2^{\alpha+1}$.

Induction: Given $2^\alpha \parallel Q_i$ and $2^{\alpha+1} \mid N - P_i^2$.

Choose r such that $2^r \parallel N - P_i^2$. Then $r > \alpha$. Let $\beta = r - \alpha$. Choose L such that $N - P_i^2 = 2^r L$. Choose M such that $Q_i = 2^\alpha M$. Since $Q_i \mid N - P_i^2$, $M \mid N - P_i^2$. But since M is odd, $\gcd(M, 2^a) = 1$, so that $M \mid L$. Let $W = \frac{L}{M}$.

Therefore, $Q_{i+1} = \frac{N-P_i^2}{Q_i} = \frac{2^r L}{2^\alpha M} = 2^{r-\alpha}\frac{L}{M} = 2^\beta W$, with W odd, so that $2^\beta \parallel Q_{i+1}$.

$$N - P_{i+1}^2 = N - (b_{i+1}Q_{i+1} - P_i)^2 = N - b_{i+1}^2 Q_{i+1}^2 + 2b_{i+1}Q_{i+1}P_i - P_i^2$$

$$= (N - P_i^2) - b_{i+1}^2 Q_{i+1}^2 + 2b_{i+1}Q_{i+1}P_i$$

$2^r \mid N - P_i^2$ and $r = \alpha + \beta > \beta$, so $2^{\beta+1} \mid N - P_i^2$.

$2^{2\beta} \mid Q_i^2$, so since $\beta > 0$, $2\beta > \beta$, so $2^{\beta+1} \mid b_{i+1}^2 Q_{i+1}^2$

$2^\beta \mid Q_i$, so $2^{\beta+1} \mid 2b_{i+1}Q_{i+1}$

Therefore, $2^{\beta+1} \mid N - P_{i+1}^2$ and the induction is complete. QED

In Theorem 2, note that if $N \equiv 1$ and N is prime, Gauss's quadratic reciprocity law states that -1 is a quadratic residue of N. Alternately, this theorem could be used as a proof of that portion of quadratic reciprocity.

Theorem 2 *If $N \equiv 1 \pmod 4$ and -1 is not a quadratic residue of N, if s is as in Lemma 5,* $\gcd(Q_s, N)$ *is a nontrivial factor of N.*

Proof: Let s be as in Lemma 5.

Case 1: m is even, choose $i = 1$. $Q_{s+1} = Q_{s-1}$. Since $Q_{s+1} = \frac{N-P_s^2}{Q_s}$ and $Q_{s-1} = \frac{N-P_{s-1}^2}{Q_s}$, this simplifies to $P_s^2 = P_{s-1}^2$, but since $\forall i\ P_i > 0$, this provides $P_s = P_{s-1}$.

But $P_s = b_sQ_s - P_{s-1} = b_sQ_s - P_s$, so $2P_s = b_sQ_s$. If $\gcd(P_s, Q_s) = 1$, $P_s \mid b_s$, so that $b_s \geq P_s$, so that $Q_s = \frac{2P_s}{b_s} \leq 2$. But this contradicts the fact that Q_s is even by Lemma 5 and $Q_s \neq 2$ by Lemma 4.

Therefore, $\gcd(P_s, Q_s) > 1$. Let $d = \gcd(P_s, Q_s)$. Then, since $N = P_i^2 + Q_iQ_{i-1}$ and $d \mid P_i$ and $d \mid Q_i$, $d \mid N$. Since also $d > 1$, d is a nontrivial factor of N.

Case 2: m is odd. Choose $i = 0$. $Q_{s+1} = Q_s$. $N = P_{s+1}^2 + Q_sQ_{s+1} = P_{s+1}^2 + Q_s^2$, so that $P_{s+1}^2 \equiv -Q_s^2 \pmod N$. If $\gcd(Q_s, N) > 1$, this is a nontrivial factor of N, and we are done. Therefore, assume that Q_s and N are relatively prime, so that $Q_s^{-1} \pmod N$ exists. Then we have $(Q_s^{-1})^2P_{s+1}^2 \equiv -1 \pmod N$. But then $Q_s^{-1}P_{s+1}$ is a square root of -1 modulo N, contradicting the fact that -1 is not a quadratic residue of N. Therefore, a nontrivial factor of N is found at s. QED

References

[C] Crandall, Richard and Carl Pomerance. *Prime Numbers: a Computational Perspective.* New York: Springer. 2001

[E] Ellis, J. H. "The history of non-secret encryption". *Cryptologia.* Number 23, July 1999, p 267-273.

[Ga] Gauss, Carl Friedrich. *Disquisitiones Arithmeticae.* Trans. by Arthur A. Clarke. New Haven, Yale University Press, 1966.

[Go] Goldstein, Larry Joel. *Abstract Algebra: A First Course.* New Jersey: Prentice-Hall, Inc. 1973. p 32-36.

[L] Lenstra, H. W. Jr. “A New Method for Factoring Integers”. Lecture Notes, 2001, University of California, Berkeley. p 5.

[P] Poorten, Alfred J. “A Note on NUCOMP”. *Mathematics of Computation.* Volume 72, Number 244, April 2003, p 1935-1946.

[Rie] Riesel, Hans. *Prime Numbers and Computer Methods for Factorization.* Boston : Birkhuser, 1985. p 191-195, 300-317.

[Riv] Rivest, R., A. Shamir and L. Adleman. “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems”. *Communications of the ACM*, 21 (2), p 120-126, February 1978

[S] Shanks, Daniel. *On Gauss and Composition II.* Mathematics Department, University of Maryland.

[T] Trappe, Wade and Lawrence C. Washington. *Introduction to Cryptography with Coding Theory.* New Jersey: Prentice-Hall, Inc. 2002. p 137-142.

[W] Williams, H. C. “Daniel Shanks (1917-1996)”. *Mathematics of Computation.* Volume 66, Number 219, July 1997, p 929-934.